6729

Network Working Group
Request for Comment # 144
NIC 6729

Arie Shoshani
SDC
April 30, 1971

Data Sharing on Computer Networks

The enclosed is an introductory paper for the meeting which will be held in Atlantic City as part of the ARPA Network meetings. The schedule for the meeting will be published soon by Steve Crocker.

The Agenda of the meeting will include:

a. Presentation of the introductory paper.

b. Open discussion to exchange comments and ideas.

c. Attempt some recommendations.

d. Possibly set up a committee of interested people.

If you have interest in the subject please plan to attend.

DATA SHARING ON COMPUTER NETWORKS

INTRODUCTION

One of the benefits expected from the use of Computer Networks is the sharing of data among users of the system. This paper is an attempt to classify the issues involved, discuss some approaches that might be taken to achieve the goal of facilitating data sharing and to point out some advantages and disadvantages of these approaches.

CONSIDERATIONS

In the process of selecting an approach one has to consider the following issues:

1. Does the approach provide the use of one language to access all data on the network?

2. Does the approach facilitate sharing of existing data created and manipulated by existing data management systems?

3. Does the approach encourage users to share data and use the facility provided? How evolutionary is the approach?

4. Could a failure of one node in the network cause the failure of the data sharing facility?

5. Does the approach promote or hinder further development of data management systems?

6. What are implementation considerations?

7. What are speed considerations?

POSSIBLE APPROACHES

1. Centralized data management system (CDMS).
   This approach is consistent with the idea that a Computer Network eventually will evolve into a collection of specialized service nodes, where each node would perform a specific function well. Users will use services on nodes according to their needs. For example, one node could be a PL/I machine (possibly a microprogrammed machine to perform PL/I compilation efficiently), another node could be a "number cruncher" for parallel-structured problems (ILLIAC IV), etc. In the same way there will be a node responsible for all data management needs for the network.

   Depending on the assumptions made one of two ways can be chosen:

   a. As assumption that we must be able to share all data, implies that the same data management system can create and manipulate this data, and therefore must perform all the functions required of a data management system, regardless of the particular use. It is generally agreed that such a task is monumental and impractical (if not impossible), since different data management systems are designed to perform specific functions well on the expense of degraded performance of other functions (e.g., fast retrieval of large files, limited updating capabilities).

   b. The assumption is made that users will share only data from the same file on a particular data management system. In this case one can implement different data management services for different tasks, but put them all on the same node to provide a data management service to the Network users. This approach can still use one

common language to access these services. This is apparently the approach taken by CCA as indicated in NIC memo 5791.

2. Standarized data management system (SDMS). In this approach a particular data management system is adopted to be implemented on all nodes. This provides for a standardized data management language as well as an identical logical data structures. Alternatively, one can choose a set of data management systems to be implemented on all nodes, then be able to share information manipulated by the same data management system on different nodes. This approach has many drawbacks as will be discussed later.

3. Integrated data management system (IDMS). This approach suggests the integration of local (to the node) data management systems and local data (files) through the use of appropriate interfaces and a common data management language.

   Under this category there may be different approaches depending on the function of the interfaces:

   a. There is an interface module in every node for every local data management system. The interface performs a dual function: on the way out--it issues requests in the common language to remote nodes; on the way in--when a request in the common language is received, the interface performs a translation from the common language to the local data management language. From a single request the translation might produce a series of commands in the local language (for example, suppose that the local language permits the specification of one quantifier only, such as "age $\leq$ 41." Suppose that the request received in the common language specifies "list all names where age $\leq$ 41 and children $\geq$5." The

translation will produce a series of commands of the form: "list all names where age ≤ 41," "save the list temporarily," "list all names in temporary file where children ≥ 5").

b. Move all local interfaces which were described above into one central node. This node is now the service node. It accepts a request in the common language and produces a series of commands to all nodes involved, in their local data management languages.

c. The local interface accepts the name of a local file (or relevant portion of the file), and sends this file to the requester after performing a translation of the data. The data can be translated using a technique such as the "Form Machine" (described in NIC 5772). The file is translated from the local data management data structure to the requesters data structure, so that the requester can perform the desired function using his local data management system.

4. Unified data management system (UDMS). This approach suggest the use of a standard interface which is to be part of every data management system on the Network. The interface has three ends. One to the user language, one to the particular physical system used and one to the Network. The interface should be global enough to permit seperation of system decisions from user language decisions. If this interface is standardized on a Network, it will facilitate communication between local data management systems in a unified way, while permitting the development and evolvement of different local data management systems. (This is a rough description of the approach taken by Barry Wesseler in Utah.)

THE COMMON LANGUAGE

It is well known that the design of a language involves a compromise between the ease of use of the language and its capability to express the functions desired. A try to merge two languages usually results in the worsening of one or both of these considerations.

For the purpose of having a common language for data management it may be desirable to seperate between the above mentioned considerations. Use natural-language for ease of use, and a formal intermediate language powerful enough to express any functions desired. This is the approach taken in the development of CONVERSE in SDC [1]. The intermediate language can be as complex as one likes since it is invisible to the user.

DISCUSSION

Predictions for future use of computers (and therfore computer networks) point out that "in 1975 we will process mostly data" [2]. Therefore, the problem of sharing data on a computer Network, as well as accessing data from remote nodes in some common language are extremely important.

If all that is desired is the sharing of data in a file by more than one user, then the CDMS approach is appropriate. Approach 1a is impractical, but 1b can provide a valuable service. Selecting this approach does not permit the sharing of existing data which was created with existing data management system, unless a restructuring of the data for the CDMS is performed. This approach does not easily permit the development of new data management systems since the CDMS should stay stable for the Network use. It does not involve translation of data or languages and therefore should provide good access speed.

The SDMS approach has many drawbacks. Selecting it implies the imposition of a particular data management system on all nodes. It inhibits further development. It does not permit the sharing of existing information. The main advantage would be the modularized structure so that the failure

of one node cannot cause the failure of the entire system. Also, because of the standardized approach sharing of data from different nodes does not involve any translation.

The main advantage of the IDMS approach is that it permits the continued use of existing data management systems with existing data bases associated with them while permitting the sharing of data among the network community of users. Since it permits the continued use of local data management systems it is the most evolutionary approach and most likely to be accepted by a user of an existing data management system. There are applications where users on each node on the Network perform mostly local access of data, and less often find it desirable to be able to share data with other nodes. For example, if hospitals are connected to nodes of a Computer Network, then most of the data about patients is accessed locally, but sometimes it is necessary to access information from other hospitals, such as global statistical information. The same situation exists for criminal files, local branches of banks, credit bureaus, warehouses, etc. Approach 3a permits the advantages of modularization, but 3b is easier to implement since no additional interfaces are necessary in the different nodes. Approach 3c seems hard to implement and can introduce inefficiencies since it involves translation from one data structure (which might be designed for efficiency) to another data structure (which may not be as sophisticated). It also involves the shipment of large amounts of data across the network.

The UDMS approach permits the continued development of local systems while facilitating a unified way for Network communication of data requests. It is not clear at this point whether this approach is practical.

Other important issues concerning sharing of data on a Computer Network, and which are mentioned in [3] are overlap of information in different files and the possibility of the same information to be contradictory, security and privacy problems, sponsors of a file vs users of a file, and others.

ACKNOWLEDGMENT

Discussions with the following people were very valuable:
Al Vorhus, Peggy Karp and others in MITRE, Barry Wesseler in Utah, Gerald Levitt, N. Cohen and others in RAND, Clark Weissman, and Charlie Kellogg in SDC, Richard Winter of CCA.

REFERENCES

1. Kellogg, C. "A Natural Language Compiler for Online Data Management." Fall Joint Computer Conference Proceedings, Vol. 33, part I, 1968. pp. 473-492

2. Clamons, Eric H. "Introductory Remarks to Data Base Management Seminar." Proceedings of Workshop on Networks of Computers (NOC-1969) NSA pp. 89-90

3. Hicken, George "Data Base Confrontation in an Information Network." Proceedings of Workshop on Networks of Computers (NOC-1969). NSA pp. 99-115.